IN THE CLAIMS

Please cancel claims 45, 47 and 49 and further amend the claims as indicated herein.

1. (Currently amended) A method for supporting versioning of data in a content management system, said method comprising:

maintaining a first table for storing an identifier of a most recent version of a data item; and

maintaining a second table for storing an identifier of an older version of said data item, wherein, when said data item is to be updated,

(i) said second table is updated to include said identifier of said most recent version of said data from said first table, so that said second table stores both of said identifier of said older version of said data item and said identifier of said most recent version of said data item, and

(ii) said first table is updated to store an identifier of a new version of said data item~~.~~, wherein a request for a version number zero (0) of said data item as part of a query, a retrieve operation, or an update operation is interpreted as a request for said data item in accordance with said identifier in said first table, and wherein a request for a version number zero (0) of said data item as part of a delete operation is interpreted as a request for said data item in accordance with an identifier of an oldest version of said data item in said second table.

2. (Previously presented) The method of claim 1, further comprising associating different version numbers with different versions of said data item.

3. (Previously presented) The method of claim 2, wherein each of said different versions is associated with a (version number − 1) value.

4. (Canceled)

5. (Previously presented) The method of claim 3, further comprising generating said (version number -1) value for successive versions of said data item by incrementing said (version number – 1) value from zero (0) to n.

6. (Previously presented) The method of claim 2, further comprising generating a value for successive versions of said data item by incrementing said version number from zero (0) to m.

7. (Canceled)

8. (Previously presented) The method of claim 1, wherein a version number having a value of zero (0) is associated with said most recent version of said data item or an oldest version of said data item, depending on a context of use for said version number.

9. (Canceled)

10. (Previously presented) The method of claim 1, wherein an operation including a version number having a value of zero (0) is interpreted as a request for said most recent version of said stored data item, and said operation is selected from a group consisting of a query operation, a retrieve operation, and an update operation.

11. (Previously presented) The method of claim 1, wherein an operation including a version number having a value of zero (0) is interpreted as a request for an oldest version of said stored data item, and said operation is a delete operation.

12. (Canceled)

13. (Original) The method of claim 1, wherein a first instance of a version of said data item is stored in said first table.

14. (Previously presented) The method of claim 1, further comprising performing a query on said first table and said second table, wherein a column attribute of a column selected by said query is retained in a result of said query.

15. (Original) The method of claim 14, wherein said query invokes a union operation.

16. (Previously presented) The method of claim 14, wherein said column attribute is obtained from a sequential query language description area (SQLDA) of said query result.

17. (Currently amended) A system for supporting versioning of data in a content management system, said system comprising:

a memory;

a module that maintains (a) a first table for storing an identifier of a most recent version of a data item in said memory, and (b) a second table for storing an identifier of an older version of said data item in said memory,

wherein, when said data item is to be updated,

(i) said second table is updated to include said identifier of said most recent version of said data from said first table, so that said second table stores both of said identifier of said older version of said data item and said identifier of said most recent version of said data item, and

(ii) said first table is updated to store an identifier of a new version of said data item, and

wherein a request for a version number zero (0) of said data item as part of a query, a retrieve operation, or an update operation is interpreted as a request for said data item in accordance with said identifier in said first table, and

wherein a request for a version number zero (0) of said data item as part of a delete operation is interpreted as a request for said data item in accordance with an identifier of an oldest version of said data item in said second table.

18. (Previously presented) The system of claim 17, further comprising a module that associates different version numbers with different versions of said data item.

19. (Previously presented) The system of claim 18, wherein each of said different versions is associated with a (version number – 1) value.

20. (Canceled)

21. (Previously presented) The system of claim 19, further comprising a module that generates said (version number -1) value for successive versions of said data item by incrementing said (version number – 1) value from zero (0) to n.

22. (Previously presented) The system of claim 18, further comprising a module that generates a value for successive versions of said data item by incrementing said version number from zero (0) to m.

23. (Canceled)

24. (Previously presented) The system of claim 17, wherein a version number having a value of zero (0) is associated with said most recent version of said data item or an oldest version of said data item, depending on a context of use for said version number.

25. (Previously presented) The system of claim 17, wherein an operation including a version number having a value of zero (0) input to said system is interpreted as a request for said most recent version of said stored data item, and said operation is selected from a group consisting of a query operation, a retrieve operation, and an update operation.

26. (Previously presented) The system of claim 17, wherein an operation including a version number having a value of zero (0) input to said system is interpreted as a request for an oldest version of said stored data item, and said operation is a delete operation.

27. (Previously presented) The system of claim 17, wherein a first instance of a version of said data item is stored in said first table.

28. (Previously presented) The system of claim 27, wherein a column attribute of a column selected by a query performed on said first table and said second table is retained in a result of said query.

29. (Original) The system of claim 28, wherein said query invokes a union operation.

30. (Previously presented) The system of claim 28, wherein said column attribute is obtained from a sequential query language description area (SQLDA) of said query result.

31. (Currently amended) A storage medium having computer readable program instructions embodied therein for supporting versioning of data in a content management system, said storage medium comprising:

    program instructions for maintaining a first table for storing an identifier of a most recent version of a data item;

    program instructions for maintaining a second table for storing an identifier of an older version of said data item ; and

    program instructions for performing an operation, wherein, when said data item is to be updated,

        (i) said second table is updated to include said identifier of said most recent version of said data from said first table, so that said second table stores both of said identifier of said older version of said data item and said identifier of said most recent version of said data item, and

        (ii) said first table is updated to store an identifier of a new version of said data item., and

wherein a request for a version number zero (0) of said data item as part of a query, a

  retrieve operation, or an update operation is interpreted as a request for said data item

  in accordance with said identifier in said first table, and

wherein a request for a version number zero (0) of said data item as part of a delete

  operation is interpreted as a request for said data item in accordance with an identifier

  of an oldest version of said data item in said second table.

32. (Previously presented) The storage medium of claim 31, further comprising program instructions for associating different version numbers with different versions of said data item.

33. (Previously presented) The storage medium of claim 32, comprising program instructions for associating each of said different versions with a (version number − 1) value.

34. (Canceled)

35. (Previously presented) The storage medium of claim 33, comprising program instructions for generating said (version number -1) value for successive versions of said data item by incrementing said (version number − 1) value from zero (0) to n.

36. (Previously presented) The storage medium of claim 32, comprising program instructions for generating a value for successive versions of said data item by incrementing said version number from zero (0) to m.

37. (Canceled)

38. (Previously presented) The storage medium of claim 31, comprising program instructions for associating a version number having a value of zero (0) with said most recent version of said stored data item or an oldest version of said stored data item, depending on a context of use for said version number.

39. (Previously presented) The storage medium of claim 31, comprising program instructions for interpreting an operation including a version number having a value of zero (0) as a request for said most recent version of said stored data item, wherein said operation is selected from a group consisting of a query operation, a retrieve operation, and an update operation.

40. (Previously presented) The storage medium of claim 31, comprising program instructions for interpreting an operation including a version number having a value of zero (0) as a request for an oldest version of said stored data item, and said operation is a delete operation.

41. (Original) The storage medium of claim 31, comprising program instructions for retaining a column attribute of a column selected by a query performed on said first table and said second table.

42. (Original) The storage medium of claim 41, wherein said query invokes a union operation.

43. (Previously presented) The method of claim 41, wherein said column attribute is obtained from a sequential query language description area (SQLDA) of said query result.

44. (Previously presented) The method of claim 1, wherein said second table stores identifiers of more than two older versions of said data item.

45. (Canceled)

46. (Previously presented) The system of claim 17, wherein said second table stores identifiers of more than two older versions of said data item.

47. (Canceled)

48. (Previously presented) The storage medium of claim 31, wherein said second table stores identifiers of more than two older versions of said data item.

49. (Canceled)